ICE2605 Signals and Systems

Course Project

Due: 8:00pm Sunday, May 29

You should submit a **single zip** file, which should include at least the following.

(1). A project report written in English and in pdf format. The report should be self-contained except when you refer to the audio clips. Include all necessary figures.

(2). All the audio clips you refer to in the report.

(3). Your source code to generate the above audio clips. You can use Python, Matlab or any other programming language you prefer.

Late policy. For late submissions, the grade will be discounted 10% per day.

Part 1 (5 points)

(1). Pick a short audio clip (e.g. music, voice) and denote its samples as $x[n]$, $n = 0, 1, \dots, N-1$. Figure out the sampling period T and reconstruct the continuous time signal $x(t)$ by some interpolation method (e.g., zero-order hold, first-order hold). Plot the waveform of $x(t)$, assuming t starts from 0. (If you use the `wav` format, the Python method `wavfile.read` of the `scipy.io` package might be useful.)

(2). Generate $x(-t)$, $x(2t)$ and $x(t/2)$, respectively, and show their waveforms.

(3). Show the spectra of $x(t)$, $x(-t)$, $x(2t)$ and $x(t/2)$, compare them and comment on their properties. Explain the assumptions and/or approximations you make.

(4). Denote by $X(j\omega)$ the spectrum of $x(t)$. Show the waveforms reconstructed using only the magnitude spectrum and the phase spectrum, respectively, i.e. $\mathcal{F}^{-1}\{|X(j\omega)|\}$ and $\mathcal{F}^{-1}\{e^{j\arg X(j\omega)}\}$. Compare them with the original signal $x(t)$.

(5). Implement an ideal low-pass filter with a cut-off frequency of your choice and apply it to $x(t)$. Show the waveform and spectrum of the output signal of the filter.

Part 2 (5 Points) For this part, you can either do the following problem, or propose your own. If you want to propose your own problem, please contact the instructor to get approval.

Speech Sampler.

(1). We can sample a CT speech signal $x(t)$ at rate f_s to produce a DT signal $x[n] = x(t/f_s)$. For an audio clip of your choice, generate different DT signals by varying the sampling rate. The recommended sampling rates are $f_{s,i} = 2^{-i} f_{s,0}$ for $i = 0, 1, 2, 3, 4$ with $f_{s,0} \geq 44$ kHz. You can try more values of i. Save the DT signals and the corresponding sampling rates as audio files. Listen to the audios and comment on their qualities. (If you use the `wav` format, the Python method `wavfile.write` of the `scipy.io` package might be useful.)

(2). Reconstruct the CT signal from the DT signals produced in (1) and visualize your reconstruction. Analyze the influence of sampling rate on the reconstruction quality and calculate the reconstruction error using a metric of your choice.

(3). Explain the basic design idea and principles of the proposed Speech Sampler.